

**BEWD CLASS 01**

# **LEARNING TO WORK LIKE A DEVELOPER**

- OR -

"How I Learned To Stop Worrying  
And Love The Command Line"

# AGENDA

- Introductions (30 minutes)
- Course Information (30 minutes)
- "What is Web Development?"
- Gaining Command of the Command Line
- Git-ting your code onto GitHub
- Practice!

# TEAM INTRODUCTIONS

Your fellow classmates are your team for the next 10 weeks.  
Let's get to know each other!

- What's your name?
- What's your favorite hobby or talent?
- Have you done any development before?
- What do you want to accomplish after finishing this course?

# **COURSE INFORMATION**

# WHAT YOU GET FROM THIS CLASS

- Guided labs in class
- Homework and extra practice
- A final project for your portfolio
- My real world experience
- Awesome new friends!

# WHAT TO EXPECT

- An intense, rigorous, difficult course
- Self-managing study time outside of class
- A fun and rewarding experience
- A valuable foundation of knowledge for future exploration and learning

# HOW TO EXCEL

- Ask questions when you have them
- Raise your hand if you have an issue
- If you get ahead, support others
- Give me feedback for what you want
- Use Schoology to ask questions to your fellow team members
- Use web search / Stack Overflow to search for answers to your questions

# PASSING THE COURSE

To pass the course you should:

- Complete 80% of all assigned homework
- Complete a final project



# REVIEW!

- Complete 80% of homework to pass
- Complete a final project to pass
- Participate and ask questions
- Very challenging and broad material
- Everyone here is on the same team!

Any questions?

# WHAT IS WEB DEVELOPMENT?

# WEB DEVELOPMENT IS CREATING WEB SITES THAT *DO SOMETHING.*

Blogs

News sites

Classified ads

Social Networks

Search Engines

Games

# 'FRONT END' AND 'BACK END': WHAT'S THE DIFFERENCE?

Front end: visual experience on the browser.  
HTML, CSS, JavaScript

Back end: browser interaction with the server  
Ruby, Java, C#.NET

Back end developers usually do the front end too.  
Often it's a placeholder for later refinement.

# FRONT END DEVELOPMENT

is like being an interior decorator



# BACK END DEVELOPMENT

is like being the architect, electrician, plumber, carpenter, foreman, and the crane driver.



# PROS AND CONS OF BACK END DEVELOPMENT ARE THE SAME

It's very complex

There is always more to learn

It never stops being challenging

There is never a shortage of work to do

# WHO'S EXCITED?

It's time to become a web developer! Are you ready?

Any questions?



**LET'S GET STARTED!**

# PART 1: LEARN THE SECRET HANDSHAKE

(Just kidding.)

# PART 1: INTRODUCING THE COMMAND LINE

aka "Terminal" aka "Shell"

This is how developers tell their computer what to do.

"Go to my home projects folder!"

"Make me a new Rails blog app!"

"Backup my code on GitHub!"

# COMMANDING THE COMMAND LINE

Our computers don't understand human language well (yet).  
We use the command line to them to tell them what to do.

"Go to my home projects folder!"

```
$ cd ~/projects
```

"Make me a new Rails blog app!"

```
$ rails new blog
```

"Backup my code on GitHub!"

```
$ git push origin master
```

# WHY USE THE COMMAND LINE?

- Efficiency
- Control
- Customization
- It's required!

Effective use of the command line  
is required to be a good developer.

# LAB EXERCISE

Learning to use the command line

1. Pair up with a teammate for paired programming.
2. Do the "ex\_command\_line\_basics.md" exercise.

Stuck? Check the cheat sheet in the "Resources.md" file.

# HOW TO STUDY

- Don't reach for the mouse.
- Do the optional homework.
- Keep practicing!

# COMMAND LINE REVIEW

This lesson is the foundation of everything you will ever do as a developer.

Any questions?



# A DEVELOPER MANTRA!

“My goal is to do the most with the least—get the maximum output with the minimum input.”

— Jarod Kintz

**PART 2:**  
**LET'S GIT ON WITH**  
**LEARNING GIT**

Learning to put your code on the cloud

# REVIEW: WHAT IS GIT?

Git is a "Version Control System"

# REVIEW: WHY USE GIT?

- Keeps your code safe
- Track code changes (that you tell it about)
- Makes your code easy to share

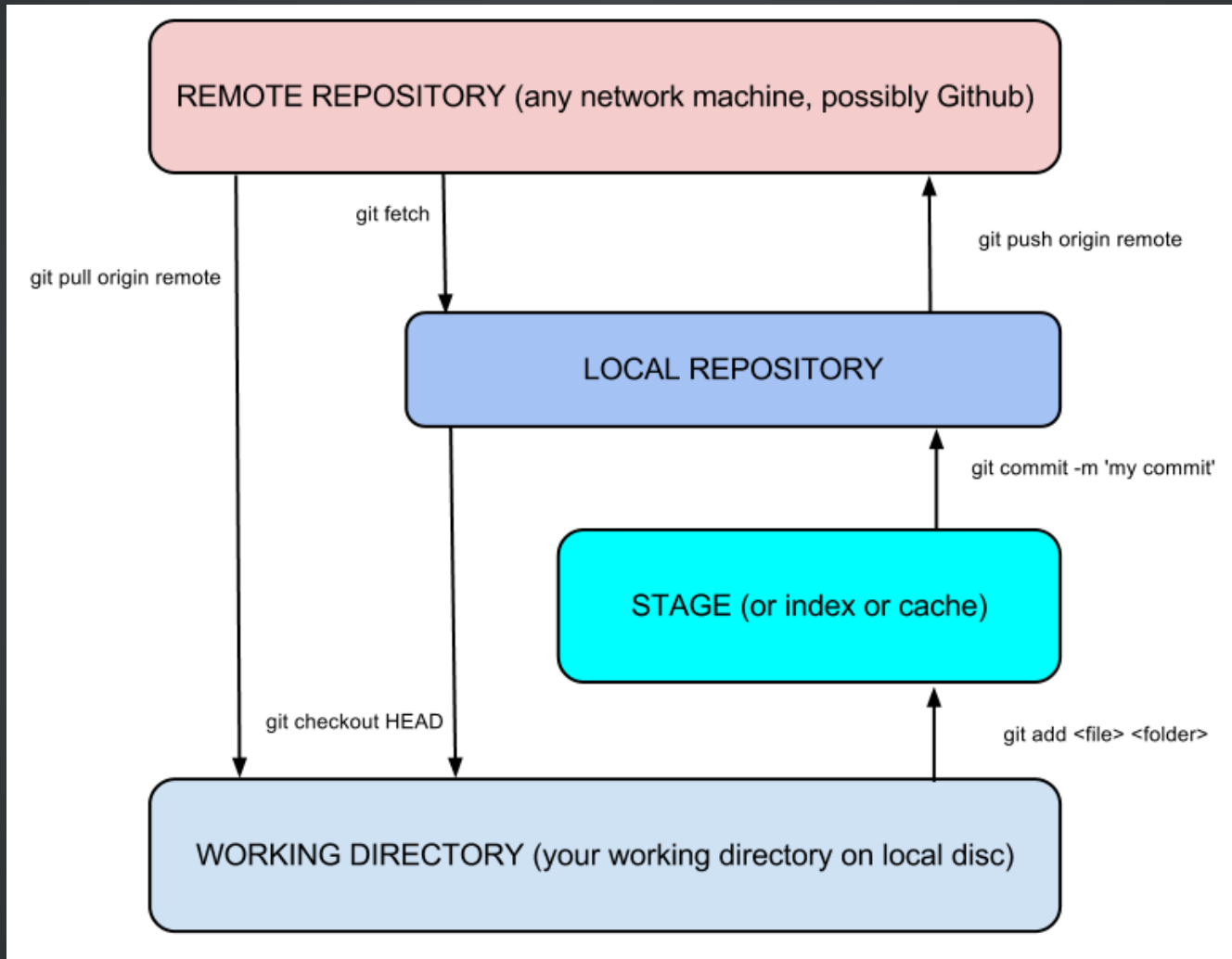
# REVIEW: WHAT IS GITHUB?

GitHub is a tool for collaborating and backing your code up online.

It is one of the most beneficial tools for development that has ever been created.

It was written in Rails!

# HOW DOES GIT WORK?



**QUESTIONS ABOUT GIT OR GITHUB?**

# LAB EXERCISE

Learning to use Git and GitHub

1. Pair up with a teammate for paired programming.
2. Do the "ex\_git\_it\_together.md" exercise.

Stuck? Check the GitHub cheat sheet link in the "Resources.md" file (near the bottom).



# HOW TO STUDY

- Study the GitHub Cheat Sheet
- Watch the Intro to Git videos
- Use GitHub!

# GIT/GITHUB REVIEW

Git is one of the most powerful and complex tools that you have available to you.

GitHub makes sharing and backing up code very easy, and makes Git easier to use.

Any questions?

# (OPTIONAL) HOMEWORK

See "Resources.md" for more study options.

## HIGHLY RECOMMENDED:

- Do "The Command Line Crash Course"!
- Watch "Intro to Git" videos.
- Review these slides again.

# END OF CLASS 1

Go team!